
Benchmarking Robust Aggregation in Decentralized Gradient Marketplaces

Zeyu Song

Department of Computer Science and Engineering
Pennsylvania State University
zysong@psu.edu

Sainyam Galhotra

Department of Computer Science
Cornell University
sg@cs.cornell.edu

Shagufta Mehnaz

Department of Computer Science and Engineering
Pennsylvania State University
smehnaz@psu.edu

Abstract

The rise of distributed and privacy-preserving machine learning has sparked interest in decentralized gradient marketplaces, where participants trade intermediate artifacts like gradients. However, existing Federated Learning (FL) benchmarks overlook critical economic and systemic factors unique to such marketplaces—cost-effectiveness, fairness to sellers, and market stability—especially when a buyer relies on a private baseline dataset for evaluation.

We introduce a comprehensive benchmark framework to holistically evaluate robust gradient aggregation methods within these buyer-baseline-reliant marketplaces. Our contributions include: (1) a simulation environment modeling marketplace dynamics with a variable buyer baseline and diverse seller distributions; (2) an evaluation methodology augmenting standard FL metrics with marketplace-centric dimensions such as Economic Efficiency, Fairness, and Selection Dynamics; (3) an in-depth empirical analysis of the existing Distributed Gradient Marketplace framework, MartFL, including the integration and comparative evaluation of adapted FLTrust and SkyMask as alternative aggregation strategies within it. This benchmark spans diverse datasets, local attacks, and Sybil attacks targeting the marketplace selection process; and (4) actionable insights into the trade-offs between model performance, robustness, cost, fairness, and stability.

This benchmark equips the community with essential tools and empirical evidence to evaluate and design more robust, equitable, and economically viable decentralized gradient marketplaces.

1 Introduction

The confluence of distributed systems and the imperative for privacy-preserving machine learning has spurred significant interest in decentralized data marketplaces Zhang et al. [2024], Li et al. [2023]. Unlike traditional centralized data brokers, these emerging ecosystems aim to facilitate data exchange directly between participants in a potentially trustless environment. A particularly promising paradigm within this space leverages Federated Learning (FL) McMahan et al. [2017] principles to enable the trading of intermediate machine learning artifacts, such as gradient updates, rather than raw data Li et al. [2023]. This approach offers enhanced privacy protection, as gradients implicitly encode learned patterns without exposing sensitive source data Geiping et al. [2020].

We focus specifically on Distributed Gradient Marketplaces (DGMs), emerging ecosystems where interactions are shaped by unique data exchange dynamics. In this scenario, multiple data owners (sellers) locally compute gradient updates—rich, privacy-preserving derivatives of their datasets—and offer these for sale. A data consumer (buyer), aiming to train a new model or enhance an existing one for a specific task, then navigates this marketplace to procure valuable gradients. A critical aspect defining this interaction, and a central theme of our investigation, is how the buyer effectively communicates their needs and ascertains the value of offered gradients. Drawing from established design patterns in decentralized data systems and specific marketplace proposals like MartFL Li et al. [2023] and DEVED Lu et al. [2024], where buyer utility is often anchored to a trusted reference, our benchmark models a scenario where the buyer utilizes their own small, private baseline dataset (D_B). This buyer-provided D_B becomes pivotal, serving a dual role: firstly, it acts as a tangible "specification-by-example," clearly conveying the buyer's desired data characteristics or target model performance, much like providing a blueprint in traditional procurement. Secondly, D_B functions as an active evaluation tool, allowing the buyer to assess the relevance, quality, and potential contribution of each incoming seller gradient. Leveraging these D_B -derived insights, the marketplace's robust aggregation mechanism then filters malicious or low-quality contributions and synergistically combines the valuable ones. This process, which can employ adapted FL techniques like FLTrust Cao et al. [2020] and SkyMask Yan et al. [2024] or MartFL's native mechanisms, is fundamentally motivated by the economic imperative for verifiable utility and targeted procurement. In a functioning marketplace, buyers must have mechanisms to ensure that the assets they acquire directly and measurably contribute to their specific objectives.

However, evaluating the efficacy and viability of such DGMs requires investigating beyond traditional FL robustness benchmarks. While standard evaluations focus on model accuracy and resilience against common attacks Bagdasaryan et al. [2020], they often neglect the unique economic and systemic considerations inherent in a decentralized market structure built around gradient exchange:

- **Economic Viability:** The cost-effectiveness for the buyer to acquire sufficient gradient quality under budget constraints Luo et al. [2021], Murhekar et al. [2023].
- **Fairness and Incentive Compatibility:** Ensuring sellers are fairly compensated based on contribution quality to sustain participation, which in turn impacts long-term market viability Yang et al. [2020], Sim et al. [2023].
- **Market Dynamics and Stability:** Understanding how the aggregator's selection patterns (e.g., diversity, consistency) influence risk concentration and market health.
- **Baseline Sensitivity:** Assessing how the characteristics of the buyer's indispensable baseline dataset D_B impact the entire marketplace's performance and fairness.
- **Market-Specific Vulnerabilities:** Defending against adversaries who may exploit the selection/payment mechanisms unique to the gradient marketplace (e.g., adversary could try to make their gradient more likely to be selected, thus impacting the fairness of the market).

Current FL benchmarks lack the metrics and scenarios to holistically evaluate these interconnected factors, hindering progress in building practical and trustworthy DGMs.

To bridge this gap, we introduce a comprehensive benchmark framework tailored for evaluating robust gradient aggregation methods operating within DGMs reliant on a buyer baseline. Figure 1 illustrates the configurable pipeline of our benchmark. Centralized Configs (e.g., YAML files) dictate the setup for each stage, influencing the Dataset Loader, the choice and parameters of Attack Plug-ins, the behavior of Seller and Buyer Simulators, and the selected Aggregator strategy. This design allows researchers to systematically vary experimental conditions. The Attack Plug-ins module enables the injection of diverse adversarial behaviors. Marketplace interactions between simulated sellers and buyers generate gradient traffic, which is then processed by the swappable Aggregator. Finally, the Metrics Logger captures a comprehensive set of model-centric and crucial marketplace-centric statistics for evaluation.

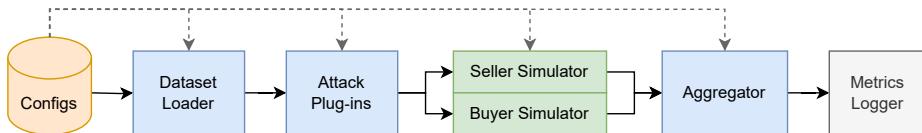


Figure 1: Overview of the Benchmark System Design.

Our main contributions through this framework include:

- ❶ A simulation environment modeling the core dynamics of DGM, including a buyer with a variable baseline D_B and sellers with data distributions controllably related to the buyer’s needs.
- ❷ A novel evaluation methodology augmenting standard FL metrics with DGM-centric dimensions: **economic efficiency**, **fairness** (incorporating gradient quality proxies), and **selection dynamics**.
- ❸ An extensive empirical benchmark centered on the MartFL Li et al. [2023] framework, which provides a comprehensive pipeline for DGM operations. Within this MartFL-based environment, we integrate and evaluate adapted versions of FLTrust Cao et al. [2020] and SkyMask Yan et al. [2024] as alternative robust aggregation strategies. This comparative evaluation spans diverse datasets and attack scenarios, including standard local attacks and coordinated Sybil attacks involving multiple malicious sellers.
- ❹ Actionable insights into the trade-offs between performance, robustness, cost, fairness, and stability when deploying these aggregation methods in the context of DGMs.

This work provides the community with crucial tools and empirical grounding for assessing technologies intended for the burgeoning field of privacy-preserving, decentralized gradient economies, ultimately aiming to foster the development of more reliable and equitable systems.

2 Related Work

Our work intersects federated learning (FL) benchmarking, robustness, data marketplaces, and fairness, with a specific focus on gradient trading reliant on a buyer’s private baseline.

Existing FL benchmarks (e.g., FedML He et al. [2020], Leaf Caldas et al. [2018], FedScale Lai et al. [2022]) primarily evaluate model accuracy and system performance under data/device heterogeneity and standard poisoning attacks like backdoors Bagdasaryan et al. [2020]. While numerous robust aggregation methods exist (e.g., Blanchard et al. [2017], Yin et al. [2018], Nguyen et al. [2022], Kabir et al. [2024]), our benchmark uniquely focuses on methods adaptable to leveraging a buyer’s trusted baseline for filtering. We evaluate FLTrust Cao et al. [2020], SkyMask Yan et al. [2024], and MartFL Li et al. [2023] (a marketplace architecture), assessing their robustness against attacks exploiting marketplace selection mechanisms, beyond standard metrics (Sections 4).

The concept of FL-based data marketplaces trading gradients is gaining traction Zhang et al. [2024]. Research explores architectures, incentive mechanisms Li et al. [2023], and cost-effectiveness Luo et al. [2021], Murhekar et al. [2023]. Fairness in FL often addresses equitable model performance Mohri et al. [2019] or traditional incentive schemes Yang et al. [2020], Zhan et al. [2020]. Our benchmark contributes by evaluating robust aggregation within this simulated marketplace context, systematically measuring novel marketplace-centric dimensions: buyer **Economic Efficiency** (cost-to-convergence); **Seller Fairness** (payment distribution, quality-reward correlation, distinguishing malicious financially); and **Market Dynamics** (selection diversity/stability). We also analyze sensitivity to buyer baseline and seller relevance, and consider related privacy risks from gradient sharing Zhu et al. [2019], Geiping et al. [2020].

Our Distinction: While prior works have proposed individual marketplace frameworks, incentive designs, or analyzed specific fairness aspects in FL, our contribution is a *comprehensive benchmarking framework* tailored to the buyer-baseline-reliant gradient marketplace scenario. This framework uniquely integrates standard FL performance and robustness evaluation with novel economic, fairness, and market dynamic metrics. It provides a structured approach to assess the practical trade-offs of deploying different robust aggregation methods in these emerging decentralized economies, especially when facing adversaries who exploit the marketplace’s operational mechanics.

3 Threat Model

We evaluate the robustness of gradient aggregation methods within the specific context of a federated data marketplace. In this setting, sellers provide gradient updates computed on their local datasets, and the buyer utilizes a small, local baseline dataset (D_B) as a reference to evaluate, select, and potentially reward these contributions via a chosen aggregation mechanism (e.g., FLTrust Cao et al. [2020], MartFL Li et al. [2023], SkyMask Yan et al. [2024] adapted for this purpose).

Our threat model assumes the presence of malicious sellers who aim to undermine the integrity or utility of the federated learning process. We assume the adversary controls a fraction of the participating sellers (varied in experiments, e.g., 10%-30%) but cannot directly compromise the buyer’s aggregation process or their baseline dataset D_B .

Adversary Capabilities: Malicious sellers possess the following capabilities:

- Data Manipulation: Full control over their local training data, enabling poisoning attacks such as embedding backdoor triggers or flipping labels.
- Gradient Computation: Ability to compute arbitrary gradient updates based on their (potentially poisoned) data or using other gradient crafting techniques.
- Protocol Knowledge: Awareness of the global model architecture and the general federated learning protocol being used.

Adversary Goals: The malicious sellers are adversarial and economically motivated:

- Model Degradation: Reduce the final global model’s accuracy on the main task (untargeted attack).
- Targeted Misbehavior: Induce specific incorrect predictions on inputs containing a backdoor trigger.
- Selection Manipulation/Unfair Gain: Craft gradients that bypass the aggregator’s filtering or selection mechanism to ensure their inclusion (and potential payment in a real marketplace), even if the gradients are non-beneficial or harmful.

The central vulnerability explored in this benchmark is the gradient selection and aggregation process reliant on the buyer’s baseline D_B . Malicious sellers seek to exploit this mechanism by submitting gradients designed to appear legitimate relative to the (potentially small or biased) D_B , thereby deceiving the aggregator. This benchmark specifically assesses how effectively different robust aggregation strategies, when adapted to use the buyer’s baseline, can mitigate these threats and ensure the integrity of the gradient-based marketplace.

4 Evaluation Methodology

Our benchmark mirrors the *end-to-end lifecycle* of a federated-gradient marketplace, from a buyer’s root dataset to post-training economic settlement. This lifecycle is divided into three macro phases, illustrated in Figure 2: **Impact of Data, Robust Training & Cost Efficiency**, and **Evaluation & Settlement**. These phases align experimental metrics, summarised in Table 1, with the real-world concerns of a gradient marketplace: (1) who is allowed in, (2) how robust and economical the training process is, and (3) how value is ultimately settled.

4.1 Pipeline Overview

The marketplace workflow is divided into three macro phases that mirror the brackets in Fig. 2.

P1 Data Impact. The buyer releases a small root dataset D_B ; methods such as FLTrust and MartFL derive a trusted *root gradient* g_B from it. A discovery mechanism then identifies candidate sellers whose private splits D_{S_i} are potentially valuable.

P2 Robust Training & Cost Efficiency. Each seller i trains locally and submits a gradient $g_{i,t}$ for the current training round t . Aggregator \mathcal{A} filters or weighs these updates, producing a selected set \mathcal{S}_t of gradients; every accepted gradient triggers a unit payment $p_{i,t} = 1$. This stage is where we log *economic* (cost), *fairness* (payment equity, benign–malicious differential), and *dynamic* (diversity, stability) statistics, alongside classical robustness metrics (accuracy, ASR, privacy leakage).

P3 Evaluation & Settlement. The aggregated gradient \bar{g}_t updates the global model. Once training ends, we report final utility, cumulative payments, and equity indicators (e.g., Gini, divergence–selection correlation), completing the settlement cycle.

4.2 Metric Taxonomy

Formal definitions, computation details, and intuition for each metric are provided in Appendix A.

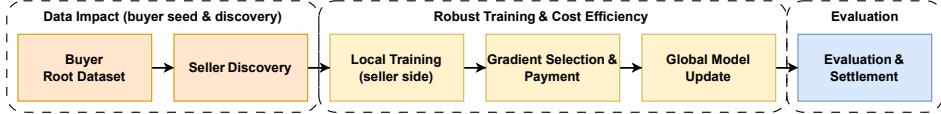


Figure 2: End-to-end pipeline with the three evaluation phases highlighted.

Table 1: Metrics aligned with the pipeline phases.

Phase	Category	Metrics
P2 Selection	Economic Fairness	Cost per Round C_t , Cost-of-Convergence CoC(\mathcal{A}) Malicious Selection Rate, Divergence–Selection Correlation, Payment Gini
P2 Dynamics	Dynamics	Selection Diversity (entropy), Stability (Jaccard)
P2–P3	Utility	Accuracy, Attack-Success Rate (ASR)
P2–P3	Privacy	Gradient-inversion quality (PSNR, SSIM)

5 Experimental Setup

Our benchmark framework supports **4 datasets**, **3 attack families**, and **4 aggregation schemes**. For detailed evaluation of our marketplace metrics, we primarily present results using the **MartFL** Li et al. [2023] aggregation scheme on the **Fashion-MNIST (FMNIST)** Xiao et al. [2017] dataset, where we perform exhaustive sweeps over adversary strength and marketplace data factors. Reproducibility scripts for all supported datasets (**CIFAR-10** Krizhevsky [2009], **AG News** Zhang et al. [2015], **TREC** Li and Roth [2002]) and aggregation schemes (**FedAvg** McMahan et al. [2017], **FLTrust** Cao et al. [2020], **SkyMask** Yan et al. [2024]) are included in the code release.

Details about the datasets and aggregation methods are presented in Appendix C. Computing resources are discussed in Appendix D.

5.1 Dataset & Model

Fashion-MNIST (FMNIST) Xiao et al. [2017]: Grayscale images 28×28 across 10 classes (60k/10k train/test). Model: LeNet-5 adapted to FMNIST. *Other* datasets (CIFAR-10, AG News, TREC) are pre-configured via `configs/*.yaml`, omitted here for brevity.

Why a Single-Dataset Deep Dive? FMNIST allows exhaustive GPU-based sweeps over adversary strength, buyer bias, and seller relevance, providing high-resolution evaluations of *accuracy*, *ASR*, *cost*, and *fairness*. The YAML-driven pipeline generalizes to all supported datasets, ensuring code breadth despite paper depth.

5.2 Marketplace Split

- **Buyer root set (\mathcal{D}_B)**: 2% of dataset. Evaluated with *unbiased* (uniform) and *biased* (Dirichlet $\alpha=0.3$) variants.
- **Seller pool (\mathcal{D}_S)**: Remaining 98%. Class distribution perturbed with multiplicative noise ($f=0.3$) around \mathcal{D}_B , simulating discovery-induced data drift.

5.3 Aggregation Schemes

We evaluate three aggregation schemes: **MartFL** Li et al. [2023], **FLTrust** Cao et al. [2020], and **SkyMask** Yan et al. [2024]. MartFL serves as the core framework, incorporating quality-aware aggregation for gradient marketplaces. We integrate adapted versions of FLTrust and SkyMask into MartFL to compare robust aggregation strategies specifically tailored to the buyer-baseline setting. These methods leverage the buyer’s root data (\mathcal{D}_B) as a trusted reference, making them compatible with our threat model.

5.4 Training Protocol

We simulate 30 sellers for 200 global rounds. In each round, a fixed fraction $p=0.3$ of sellers are sampled for 2 local epochs. Local updates use Adam Kingma [2014] (batch size 64, LR 0.001; constant LR). Each configuration is repeated 10 times, reporting mean results.

5.5 Attack Suite

We evaluate robustness against two categories of attacks: **Local Attacks**, executed independently, and **Sybil Attacks**, involving colluding malicious sellers. Our benchmark framework implements several attack techniques, including **Backdoor** (e.g., sample modification with triggers and relabeling), **Label Flipping** (randomly flipping labels), and **Adaptive Mimicry** (a Sybil technique blending gradients to exploit selection criteria). These attack implementations feature various tunable parameters (e.g., poisoning rate, trigger size/location, blending weights, label flip percentage, adversarial fraction, etc.) to enable flexible evaluation of defenses.

In this paper, we primarily focus on demonstrating the benchmark’s use case through a detailed analysis of the **Backdoor** attack in both Local and Sybil scenarios. The specific parameters used for these analyses include a fraction (10% and 50%) of malicious sellers’ local samples modified, and the fraction of adversarial sellers in the network varying among $\{0.2, 0.3, 0.4\}$. Detailed backdoor setup is provided in Appendix B.

6 Results and Discussion

6.1 Deep-Dive: MartFL

6.1.1 Robustness vs. Integrity Attacks

Figure 3a starkly exposes a classic *stealthy failure* inherent in the system. As the fraction of malicious sellers (Adversary Rate) doubles from 20% to 40%, the main task accuracy on clean data ‘Main Acc. (w/ backdoor)’ exhibits remarkable resilience, slipping by a mere 1.5 percentage points (from approximately 0.855 to 0.84). This performance is only marginally lower than the ‘Main Acc. (no attack)’ baseline of 0.86, potentially masking the underlying compromise. However, this perceived robustness belies a critical vulnerability: the backdoor Attack Success Rate (ASR) (“Backdoor ASR”) soars from 0.63 to 0.86. This 23 percentage point surge in ASR represents a near-linear growth in vulnerability with the adversary rate, achieving high attack efficacy with almost no perceptible loss in utility. The mechanism enabling this stealthy compromise is elucidated by the selection dynamics in Figure 3b. This figure reveals that MartFL’s similarity-based filter, while showing some efficacy, ultimately permits a sufficient volume of malicious gradients to implant the backdoor. For instance, attackers employing a “Standard Backdoor” still see their updates selected between 14.5% (at 20% adversary rate) and 30.5% (at 40% adversary rate) of the time. While this is consistently lower than the “Control Group” (benign sellers, selected 19.5% to 36% of the time), indicating some filtering capability, the selection rate is clearly adequate for successful poisoning. The vulnerability is exacerbated by more sophisticated adversaries. Notably, under the “Sybil Backdoor” attack, adversaries achieve an even higher selection rate, ranging from approximately 20% to 31%. This rate is remarkably close to that of the “Control Group”, and even slightly surpasses it at the 20% adversary rate. This suggests that Sybil attackers, by mimicking the gradient characteristics of benign or previously selected sellers, can largely circumvent MartFL’s selection mechanism. The filter struggles to discriminate these more advanced malicious updates, increasing the likelihood that their harmful gradients are incorporated into the global model, thus directly contributing to the high ASR observed in Figure 3a with minimal impact on main accuracy. This underscores a critical challenge: defenses based solely on simple similarity may be insufficient against adaptive adversaries capable of strategic mimicry.

6.1.2 Cost & Convergence

Figures 3c and 3d reveal the economic and temporal implications for a buyer aiming to achieve specific accuracy milestones (70%, 80%, and 85%) under different attack scenarios.

Under the No Attack baseline, reaching the 85% accuracy milestone requires the buyer to purchase approximately 460 gradients, achieved over roughly 22 communication rounds. When subjected to a Standard Backdoor attack, there's a mixed impact on efficiency from the buyer's perspective. While the number of gradients required to reach 85% accuracy decreases to around 400 (a 13% cost saving for the buyer), the convergence time extends to approximately 24.5 rounds (an 11% increase in time). This scenario presents a trade-off: reduced gradient expenditure at the cost of slower model training.

Most strikingly, the Sybil Backdoor variant presents a counterintuitive outcome. To reach the 85% accuracy milestone, this attack scenario requires only about 355 gradients—a substantial 23% reduction in cost compared to the No Attack baseline and 11% less than the Standard Backdoor. Furthermore, it achieves this milestone in approximately 21.5 rounds, which is marginally faster than the No Attack scenario and notably quicker than the Standard Backdoor. This implies that, from a purely cost-and-speed perspective for achieving main task accuracy, the Sybil-attacked environment appears more efficient for the buyer.

These findings critically underscore that convergence speed and gradient cost, when evaluated solely on main task accuracy, are unreliable and potentially misleading indicators of malicious activity. The Sybil attack, in particular, demonstrates an ability to embed a backdoor (as seen in previous ASR figures) while simultaneously appearing to enhance the economic and temporal efficiency for the buyer. This "deceptive efficiency" could occur if the Sybil attackers, by mimicking high-quality updates (as suggested by their higher selection rates in Figure 3b), contribute gradients that are effective for the main task in addition to carrying the backdoor payload.

Therefore, a buyer relying solely on these superficial efficiency metrics might inadvertently favor a system compromised by a sophisticated Sybil attack, acquiring a performant yet backdoored model at a lower apparent cost. This highlights the vulnerability of marketplaces if buyers optimize for simple cost/speed metrics without robust integrity verification, and it demonstrates that attackers can manipulate these dynamics to their advantage without raising immediate red flags based on common performance indicators. The idea that a malicious seller could inflate costs is valid, but these results show a more insidious scenario where certain attacks reduce the buyer's perceived cost to reach a target accuracy, making the attack even harder to detect through economic signals alone.

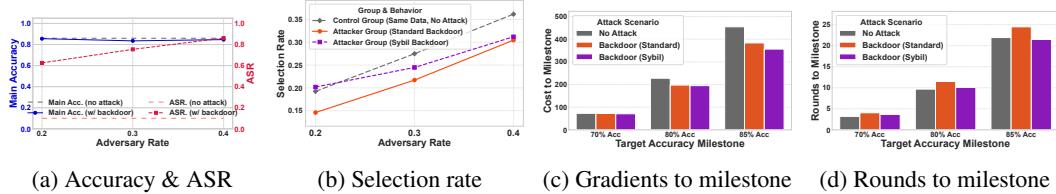


Figure 3: MartFL performance across adversary rates: accuracy, attack success, filtering robustness, and convergence speed/cost.

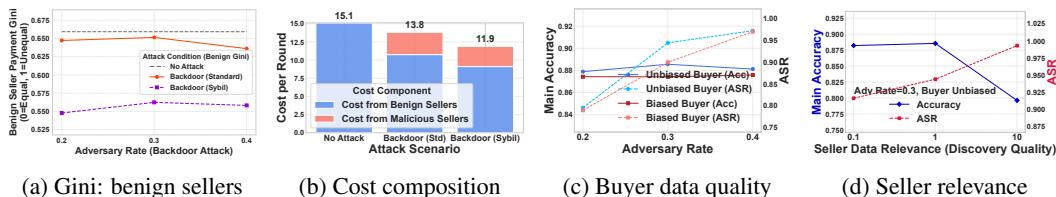


Figure 4: (a–b) Seller-centric analysis under backdoor attacks: Gini coefficient of benign selection (a) and average cost breakdown (b). (c–d) Robustness sensitivity to (c) buyer data quality and (d) seller relevance in FMNIST.

6.1.3 Marketplace Fairness: Economic Disadvantage for Benign Sellers

The introduction of adversaries into the marketplace reveals significant fairness concerns, particularly impacting benign sellers, as illustrated in Figures 4a and 4b.

Figure 4a evaluates the Gini coefficient for payments received by benign sellers. Under a Standard Backdoor attack, the Gini coefficient remains virtually unchanged from the No Attack baseline

(approx. 0.65 vs. 0.66 at a 0.3 adversary rate). This suggests that this type of attack, while compromising model integrity, does not fundamentally alter the relative reward distribution among benign sellers who are selected.

In stark contrast, the **Sybil Backdoor Attack** leads to a notable decrease in the Gini coefficient for benign sellers (e.g., to approximately 0.56 at a 0.3 adversary rate). This indicates that payments among the selected benign sellers become more evenly distributed. However, this apparent "improvement" in equality is deceptive, as it stems from malicious Sybil attackers successfully mimicking high-quality gradients. By doing so, they are selected more often (as seen in Figure 3b), effectively siphoning off selections that would have otherwise gone to the most competitive (and thus higher-earning) benign participants. This curtails the earnings of top-performing benign sellers, leading to a flatter, more "equal" distribution among a smaller, less rewarded pool of selected benign participants. Figure 4b (examining an adversary rate of 0.3) provides a clearer picture of the economic toll. In the **No Attack** scenario, the average cost per round (representing selected gradients paid for by the buyer) is approximately 15.1, entirely attributed to benign sellers.

Under the **Standard Backdoor Attack**, the total cost per round drops to 13.8. Crucially, the cost attributed to benign sellers plummets to 10.8 (a 28.5% decrease from the **No Attack** scenario), while malicious sellers now account for 3.0 of the cost. The situation worsens with the **Sybil Backdoor Attack**. The total cost per round falls further to 11.9. Here, benign sellers' contribution drops to just 9.1 (a staggering 39.7% decrease from **No Attack**), while malicious sellers contribute 2.8. While the absolute cost from malicious sellers slightly decreases from **Standard** (3.0) to **Sybil** (2.8) in this snapshot, their proportion of the (reduced) total selections increases from 0% (**No Attack**) to 21.7% (**Standard**) and then to 23.5% (**Sybil**). More importantly, the overall reduction in selections per round (from 15.1 to 11.9) disproportionately impacts benign sellers, whose collective remuneration is significantly diminished. The marketplace's similarity-based filter, therefore, fails to protect honest sellers' revenue; instead, it inadvertently penalizes them while still allowing malicious updates (especially sophisticated Sybil ones) to pass through.

Taken together, these results expose a critical vulnerability: MartFL, in its current form, allows attackers to not only compromise model integrity but also to significantly undermine the economic fairness for benign participants. The **Sybil Attack** is particularly pernicious, as it makes the reward distribution among remaining benign sellers appear more equitable (lower Gini) while simultaneously reducing their overall income and market share. This highlights that similarity-only screening is insufficient, necessitating mechanisms like provenance tracking or reputation systems to maintain both economic fairness and model security.

6.1.4 Sensitivity of MartFL Robustness to Marketplace Data Factors

This section investigates the robustness of MartFL under varying marketplace data conditions, specifically focusing on the impact of the buyer's initial data distribution (buyer-root bias) and the relevance of data provided by sellers (discovery quality). These factors are critical as they reflect realistic variations in a dynamic data marketplace.

Impact of Buyer-Root Bias. Figure 4c compares MartFL's performance with *unbiased* versus *biased* buyer-root datasets (Dirichlet $\alpha=0.3$). However, Attack Success Rates (ASR) are critically high for both, escalating sharply with adversary presence. For instance, unbiased root ASRs range from 0.79 (20% adversaries) to 0.97 (40%), while biased root ASRs, though occasionally slightly lower (e.g., 0.90 vs. 0.95 at 30% adversaries), also remain excessively high (often $>85\%$). These consistently high ASRs, despite minor variations, indicate that initial buyer-root bias minimally impacts backdoor vulnerability. This is because MartFL's dynamic reference centroid, shaped by selected seller data, dilutes the buyer's initial data characteristics. Consequently, trust shifts to the seller selection process, which remains exploitable if malicious sellers mimic the evolving centroid, particularly if they contribute to the reference set.

Impact of Seller-Data Relevance (Discovery Quality). Figure 4d illustrates the system's sensitivity to the relevance or quality of data provided by sellers, with a fixed adversary fraction of 30% and an unbiased buyer. Here, "Seller Data Relevance" values from 0.1 (high relevance/low noise) to 10 (low relevance/high noise/heterogeneity) are examined.

With highly relevant seller data (relevance 0.1), the main accuracy is 0.882, and ASR is already high at 0.925. A slight, non-critical accuracy peak occurs at relevance ‘1’ (0.885), a common deep learning behavior. However, the dominant trend is clear: as data relevance degrades towards ‘10’, main accuracy plummets to 0.795, while ASR climbs to nearly 0.99. This reveals a critical vulnerability: poorer or more heterogeneous data hampers model utility and drastically increases backdoor susceptibility. MartFL’s similarity-based filter struggles with diverse data, finding it harder to distinguish benign heterogeneity from malicious poisoning. Consequently, increased data heterogeneity broadens the attack surface, easing unfiltered backdoor injection.

Implication for Marketplace Defenses. The findings from the buyer and seller data analyses underscore that metrics and defenses tied solely to data similarity (like gradient proximity) are insufficient for robust security in a diverse data marketplace. MartFL’s susceptibility to high ASR, irrespective of buyer bias, and its exacerbated vulnerability with decreasing data relevance highlight this point. Future marketplace defenses must urgently incorporate more sophisticated mechanisms, such as provenance tracking, historical reputation systems, or robust anomaly detection techniques that go beyond simple similarity checks, to ensure both model integrity and fair participation.

6.2 Comparative Filtering Performance of Aggregation Methods

We benchmarked the filtering capabilities of FLTrust and SkyMask alongside MartFL within our DGM simulation to assess their ability to mitigate malicious gradient submissions. Table 2 presents Malicious Selection Rate (MSR) against a backdoor attack (30% malicious sellers, 30% local data poisoned), compared to benign seller rates under no attack (BSR). Experiments used LeNet-5 (FMNIST), a CNN (CIFAR-10), and TextCNN (TREC). Originally designed for image datasets, SkyMask was tested in our work on such data only, aligning with its initial implementation. The results highlight that these similarity-based methods exhibit varied and often limited filtering efficacy against backdoor attacks in the DGM context. FLTrust generally shows the most consistent, though modest, reduction in malicious seller selection (e.g., CIFAR-10 MSR: 0.27 vs. BSR: 0.30). MartFL’s performance is inconsistent across datasets, notably struggling on CIFAR-10 (MSR: 0.46 vs. BSR: 0.29). SkyMask, on image datasets, demonstrates minimal filtering, with malicious sellers selected nearly as often as benign ones. This comparative overview suggests that current robust aggregation techniques, when applied to DGMs, can be frequently bypassed by backdoor attacks, underscoring the need for more robust defenses

Table 2: Comparative filtering performance.

Dataset	MartFL		FLTrust		SkyMask	
	BSR	MSR	BSR	MSR	BSR	MSR
FMNIST	0.29	0.18	0.30	0.29	0.30	0.30
CIFAR-10	0.29	0.46	0.30	0.27	0.30	0.30
TREC	0.29	0.24	0.30	0.24	–	–

7 Conclusion

We present the first benchmark tailored to *buyer-baseline* Distributed Gradient Marketplaces (DGMs). Our modular simulator, enriched with economic, fairness, and selection-dynamics metrics, exposes how market-aware attacks defeat three state-of-the-art aggregators (MartFL, FLTrust, SkyMask) even when classical accuracy appears unchanged. The benchmark, therefore, offers an essential testbed—and reproducible code—for building robust and equitable gradient economies.

Limitations. Our current scenarios focus on poisoning-style threats and simplified seller interactions; we do not yet capture large-scale, rapidly evolving markets or the full spectrum of strategic behaviours. Furthermore, while our benchmark framework supports a broader range of datasets and aggregation schemes (as detailed in Appendix C and available in our code release), the main paper presents an in-depth analysis primarily on a subset to ensure clarity and focus.

Future Work. (1) Integrate orthogonal trust signals (reputation, provenance); (2) Design multi-stage filters and incentive-compatible rewards; (3) Extend to adaptive, multi-round adversaries and larger market scales.

Broader Impacts. By standardizing robustness and fairness evaluation, the benchmark can guide the creation of safer, more inclusive DGMs. At the same time, public release may help adversaries tune stronger attacks; we therefore encourage paired research on complementary defenses and responsible deployment practices (e.g., provenance tracking, gated access).

References

- Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *International conference on artificial intelligence and statistics*, pages 2938–2948. PMLR, 2020.
- Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/f4b9ec30ad9f68f89b29639786cb62ef-Paper.pdf.
- Sebastian Caldas, Jakub Konečný, H Brendan McMahan, and Ameet Talwalkar. Leaf: A benchmark for federated settings. In *Workshop on Federated Learning and Analytics (NeurIPS 2018)*, 2018. arXiv preprint arXiv:1812.01097.
- Xiaoyu Cao, Minghong Fang, Jia Liu, and Neil Zhenqiang Gong. Fltrust: Byzantine-robust federated learning via trust bootstrapping. *arXiv preprint arXiv:2012.13995*, 2020.
- Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients—how easy is it to break privacy in federated learning? *Advances in neural information processing systems*, 33:16937–16947, 2020.
- Chaoyang He, Songze Li, Jinhyun So, Mi Zhang, Hongyi Wang, Xiaoyang Wang, Praneeth Vepakomma, Abhishek Singh, Hang Qiu, Li Shen, et al. Fedml: A research library and benchmark for federated machine learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 1–17, 2020. Also available as arXiv preprint arXiv:2007.13518.
- Ehsanul Kabir, Zeyu Song, Md Rafi Ur Rashid, and Shagufta Mehnaz. Flshield: a validation based federated learning framework to defend against poisoning attacks. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 2572–2590. IEEE, 2024.
- Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. URL <https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf>.
- Fan Lai, Yinwei Dai, Sanjay Singapuram, Jiachen Liu, Xiangfeng Zhu, Harsha Madhyastha, and Mosharaf Chowdhury. Fedscale: Benchmarking model and system performance of federated learning at scale. In *Proceedings of the 39th International Conference on Machine Learning*, pages 11814–11827. PMLR, 2022.
- Qi Li, Zhuotao Liu, Qi Li, and Ke Xu. martfl: Enabling utility-driven data marketplace with a robust and verifiable federated learning architecture. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 1496–1510, 2023.
- Xin Li and Dan Roth. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*, 2002.
- Charles Lu, Baihe Huang, Sai Praneeth Karimireddy, Praneeth Vepakomma, Michael Jordan, and Ramesh Raskar. Daved: Data acquisition via experimental design for data markets. *arXiv preprint arXiv:2403.13893*, 2024.
- Bing Luo, Xiang Li, Shiqiang Wang, Jianwei Huang, and Leandros Tassiulas. Cost-effective federated learning design. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, pages 1–10. IEEE, 2021.

- Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.
- Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97, pages 4615–4624. PMLR, 2019.
- Aniket Murhekar, Zhuowen Yuan, Bhaskar Ray Chaudhury, Bo Li, and Ruta Mehta. Incentives in federated learning: Equilibria, dynamics, and mechanisms for welfare maximization. *Advances in Neural Information Processing Systems*, 36:17811–17831, 2023.
- Thien Duc Nguyen, Phillip Rieger, Huili Chen, Hossein Yalame, Helen Möllering, Hossein Fereidooni, Samuel Marchal, Markus Miettinen, Azalia Mirhoseini, Shaza Zeitouni, et al. {FLAME}: Taming backdoors in federated learning. In *31st USENIX security symposium (USENIX Security 22)*, pages 1415–1432, 2022.
- Rachael Sim, Yehong Zhang, Nghia Hoang, Xinyi Xu, Bryan Kian Hsiang Low, and Patrick Jaillet. Incentives in private collaborative machine learning. *Advances in Neural Information Processing Systems*, 36:7555–7593, 2023.
- Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017. URL <https://arxiv.org/abs/1708.07747>.
- Peishen Yan, Hao Wang, Tao Song, Yang Hua, Ruhui Ma, Ningxin Hu, Mohammad Reza Haghighat, and Haibing Guan. Skymask: Attack-agnostic robust federated learning with fine-grained learnable masks. In *European Conference on Computer Vision*, pages 291–308. Springer, 2024.
- Jiachen Yang, Ang Li, Mehrdad Farajtabar, Peter Sunehag, Edward Hughes, and Hongyuan Zha. Learning to incentivize other learning agents. *Advances in Neural Information Processing Systems*, 33:15208–15219, 2020.
- Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International conference on machine learning*, pages 5650–5659. Pmlr, 2018.
- Yae Zhan, Peng Li, and Yong Qu. A learning-based incentive mechanism for federated learning. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119, pages 11308–11317. PMLR, 2020.
- Jiayao Zhang, Yuran Bi, Mengye Cheng, Jinfei Liu, Kui Ren, Qiheng Sun, Yihang Wu, Yang Cao, Raul Castro Fernandez, Haifeng Xu, Ruoxi Jia, Yongchan Kwon, Jian Pei, Jiachen T. Wang, Haocheng Xia, Li Xiong, Xiaohui Yu, and James Zou. A survey on data markets, 2024. URL <https://arxiv.org/abs/2411.07267>.
- Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28, 2015.
- Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer **[Yes]**, **[No]**, or **[NA]**.
- **[NA]** means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "**[Yes]**" is generally preferable to "**[No]**", it is perfectly acceptable to answer "**[No]**" provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "**[No]**" or "**[NA]**" is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer **[Yes]** to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading “NeurIPS Paper Checklist”.**
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers.**

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

Answer: **[Yes]**

Justification: The abstract and introduction accurately reflect the paper’s contributions and scope. Specifically, we benchmark the distributed gradient marketplace’s robustness.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: **[Yes]**

Justification: We discuss the limitation in the Section 7.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We will submit our code.

Guidelines:

- The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code used for our experiments is available.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We will submit our code.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We mentioned the statistical significance in Section 5. We will submit our code.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Appendix D.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: NA.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Section 7.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: NA

Guidelines:

- The answer NA means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: MIT LICENSE for our code.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: In the readme of our submitted code.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: NA

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: NA.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: NA.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

A Metric Definitions and Computation Details

Table 3 expands Table 1 with exact formulas, normalisation choices, and implementation notes.

B Backdoor Attack Details

Our benchmark implements backdoor attacks adaptable to different data modalities. The core idea is to associate a specific, often innocuous, trigger pattern with a target label, such that models trained on backdoored data will misclassify inputs containing the trigger.

For image datasets, such as Fashion-MNIST and CIFAR-10, the backdoor is typically implemented by embedding a visual trigger. In our experiments, we use a 10×10 white pixel patch placed in

Table 3: Formal definitions of benchmark metrics. S_t is the set of selected sellers in round t , g_i a gradient, and $f(\cdot)$ the task loss.

Metric	Definition / Computation
Cost per Round C_t	Number of gradients selected in round t (unit payment model).
Cost-of-Convergence $\text{CoC}(\mathcal{A})$	$\sum_{t=1}^{T^*} C_t$, where T^* is the first round reaching a target accuracy.
Malicious Selection Rate (MSR)	$\frac{\sum_{t=1}^T S_t \cap \mathcal{M} }{\sum_{t=1}^T S_t }$. Fraction of all selected gradients that come from malicious sellers.
Divergence-Selection Corr.	Pearson correlation between $\ g_i - g_B\ _2$ and a binary variable indicating selection.
Payment Gini	Gini coefficient over cumulative payments to benign sellers.
Selection Diversity	Shannon entropy of the empirical distribution of seller IDs in $\{S_t\}_{t=1}^T$.
Stability (Jaccard)	$\frac{1}{T-1} \sum_{t=1}^{T-1} \frac{ S_t \cap S_{t+1} }{ S_t \cup S_{t+1} }$.
Accuracy	Standard top-1 accuracy on a held-out clean test set.
Attack-Success Rate (ASR)	Fraction of triggered inputs classified into the target label.
Gradient-Inversion PSNR / SSIM	Peak-Signal-to-Noise Ratio and Structural Similarity between reconstructed and ground-truth inputs (details follow).

the bottom-right corner of the image, as illustrated in Figure 5. Images containing this trigger are relabeled to a specific target class (e.g., class 0).

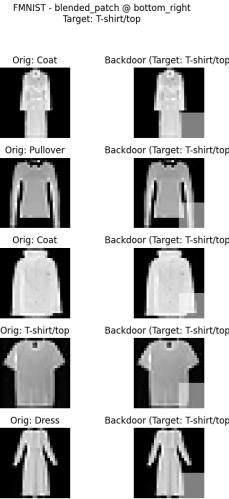


Figure 5: Example of a backdoor trigger applied to Fashion-MNIST. A 10×10 white patch is embedded in the bottom-right corner.

For text datasets, such as AG News or TREC, a textual trigger is used. This often involves inserting a specific rare word or a sequence of tokens that would not naturally occur in clean samples. In our setup, for example, we might insert a specific rare token (e.g., "cf") or a predefined sequence at the end of backdoored text samples, which are then relabeled to a target category.

The fraction of samples within a malicious seller’s local dataset that are backdoored, the specific trigger pattern, its location (for images), and the target label are all configurable parameters within our benchmark framework. The examples shown here represent the specific configurations used for the results presented in the main paper.

C Dataset and Aggregation Methods

In this section, we detail the datasets used for evaluating the performance of various federated learning aggregation strategies and briefly describe the key characteristics of each aggregation method compared in our experiments.

C.1 Datasets

We selected a diverse set of benchmark datasets covering both image and text classification tasks to thoroughly evaluate the aggregation methods under different data characteristics and complexities.

Fashion-MNIST (FMNIST): Xiao et al. [2017] This is a dataset of grayscale images of clothing items from 10 categories. It serves as a direct drop-in replacement for the original MNIST dataset but is generally considered more challenging. It consists of 70,000 images, split into 60,000 training and 10,000 testing examples. Each image is 28×28 pixels.

CIFAR-10: Krizhevsky [2009] A widely used dataset for image classification, consisting of 60,000 32×32 color images in 10 classes. There are 50,000 training images and 10,000 test images. The classes represent common objects and animals (e.g., airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck).

TREC-6: Li and Roth [2002] A dataset for question type classification. It consists of questions categorized into 6 coarse types: Abbreviation (ABBR), Entity (ENTY), Description (DESC), Human (HUM), Location (LOC), and Numeric (NUM). It is a relatively small text dataset, commonly used for benchmarking text classification models, typically with around 5,500 training examples and 500 test examples.

AG News: Zhang et al. [2015] A large text classification dataset derived from the AG corpus. It contains news articles categorized into 4 classes: World, Sports, Business, and Sci/Tech. It comprises 120,000 training samples and 7,600 test samples, providing a larger-scale text classification benchmark compared to TREC-10.

C.2 Aggregation Methods

We evaluate the performance of several prominent aggregation strategies designed for federated learning, representing different approaches to combining client model updates.

FedAvg (Federated Averaging). McMahan et al. [2017] As the foundational and most widely used algorithm in federated learning, FedAvg averages the model weights (or gradients) received from selected clients, weighted by the number of data samples each client holds. It serves as the primary baseline for comparison.

SkyMask. Yan et al. [2024] As a novel attack-agnostic robust federated learning defense, SkyMask addresses the challenge of fine-grained Byzantine attacks which can bypass existing defenses. Its key innovation is the use of fine-grained learnable masks applied at the parameter level. These masks are trained on a small server-side root dataset to identify and mitigate the impact of malicious model updates, providing robustness against a wide range of attacks.

MartFL. Li et al. [2023] This method is presented as part of a novel federated learning architecture specifically designed for secure, utility-driven data marketplaces. Focusing on robustness against malicious clients and addressing challenges in quality assessment, MartFL employs a quality-aware model aggregation protocol. This protocol enables the server (acting as a data acquirer) to evaluate and selectively aggregate client updates based on their quality, effectively excluding low-quality or poisoned models, even when using a potentially biased server-side dataset.

FLTrust. Cao et al. [2020] As a Byzantine-robust federated learning method, FLTrust introduces the concept of the server bootstrapping trust. It achieves this by having the server maintain a model based on a small, clean root dataset. In each round, the server assesses the trustworthiness of client model updates by comparing their direction and magnitude similarity to the server model update. Client updates are assigned a trust score (using ReLU-clipped cosine similarity) and normalized in magnitude before being aggregated into the global model. This trust-based aggregation strategy provides resilience against malicious clients and various poisoning attacks.

D Computing Resources

Experiments were conducted on a server equipped with dual Intel(R) Xeon(R) Gold 5317 CPUs (24 cores/48 threads total), 503 GiB of RAM, and a pool of eight NVIDIA RTX A6000 GPUs, each with 48 GB of VRAM. The system utilized CUDA 12.4 and NVIDIA driver version 550.90.07, running an Ubuntu-based Linux distribution.

For the results presented in this paper, individual experimental runs typically utilized a single NVIDIA RTX A6000 GPU. The total runtime for experiments varied based on hyperparameters such as the number of clients in federated learning, dataset complexity, model architecture, and the aggregation mechanism. As a representative example, training LeNet-5 on the FMNIST dataset for 30 clients and 200 global rounds took approximately 6 hours on a single NVIDIA RTX A6000 GPU.